



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

Address: COMMISSIONER FOR PATENTS

P.O. Box 1450

Alexandria, Virginia 22313-1450

www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/763,866	01/22/2004	Edward Eytchison	Sony-29000	8507
28960 7590 05/13/2010 HAVERSTOCK & OWENS LLP 162 N WOLFE ROAD SUNNYVALE, CA 94086				
EXAMINER ALL FARIAD				
ART UNIT 2446		PAPER NUMBER		
MAIL DATE 05/13/2010		DELIVERY MODE PAPER		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents  
United States Patent and Trademark Office  
P.O. Box 1450  
Alexandria, VA 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

**BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES**

Application Number: 10/763,866  
Filing Date: January 22, 2004  
Appellant(s): EYTCHISON ET AL.

\_\_\_\_\_  
Jonathan Owens  
(Registration # 37,902)  
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed 03/07/2010 appealing from the Office action mailed 06/22/2009.

**(1) Real Party in Interest**

A statement identifying by name the real party in interest is contained in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The following is a list of claims that are rejected and pending in the application:

Claims 1, 2, 4-11, 13, 15-20 and 22-24 are rejected and pending.

**(4) Status of Amendments After Final**

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

**(5) Summary of Claimed Subject Matter**

The examiner has no comment on the summary of claimed subject matter contained in the brief.

**(6) Grounds of Rejection to be Reviewed on Appeal**

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

**(7) Claims Appendix**

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

**(8) Evidence Relied Upon**

US 2003/0204612	Warren, Mark	10-2003
US 5,623,695	Lozinski et al.	04-1997

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1, 2, 4-11, 13, 15-20, and 22-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Warren (US 2003/0204612 A1) in view of Lozinski (US 5,623,695 A).

### **Claim 1**

Warren teaches a method comprising:

searching for at least one device based on a content type (Paragraph [0024] "A web services protocol is a protocol that may allow a web service to be published, located, and invoked over a network, although any other and/or additional functions may be supported by the web services protocol" and paragraph [0053] In one embodiment, database 236 stores device information 254. In a particular embodiment, device information 254 identifies each network element 108 in system 100 and the device type of each network element 108. For example, each network element 108 may communicate using at least one communications protocol, and network elements 108 may be divided into groups or device types based on the communications protocol used by network elements 108);

detecting the at least one device (See figure 4, number 404 "identify network element");

detecting a protocol associated with each device (See figure 4, number 406 "identify communications protocol");

matching the detected protocol with a protocol translator module; and using the protocol translator module to translate a command formatted in the protocol into a translated command (See figure 4, number 414 "translate device command" and paragraph [0020] "In a particular embodiment, manager 102 communicates using a web services protocol, and abstraction device 106 translates between the web services protocol and the protocols used by network elements 108. This may allow manager 102 to communicate with different network elements 108 using a common protocol").

Warren fails to teach wherein the translated command is formatted in a common application programming interface, wherein the common application programming interface is a single application programming interface that is configured to be used by a plurality of applications.

However, Lozinski teaches in Column 1 lines 50-54 "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" in order that "an application program can select at any particular time which of the facilities to communicate with using the interface" (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" as taught by Lozinski in order that "an application program can select at any particular time which of the facilities to communicate with using the interface".

**Claim 2**

The modified Warren teaches the method according to claim 1, further comprising searching for the device from a plurality of devices based on a device identifier (paragraph [0065] "Device identifier").

**Claim 4**

The modified Warren teaches the method according to claim 1, further comprising searching for the device from a plurality of devices based on a device type (paragraph [00703] "Device type identifier").

**Claim 5**

The modified Warren teaches the method according to claim 1, further comprising searching for the device from a plurality of devices based on a device's availability (paragraph [0065] "Device identifier" and "Other and/or additional information may be included in identification information 352 without departing from the scope of the present invention").

**Claim 6**

The modified Warren teaches the method according to claim 1, further comprising searching for the protocol translator module (paragraph [0070] "Abstraction device 206 identifies the communications protocol used by the identified network

element 108 at step 406. This may include, for example, command translator 234 using device information 254 and/or device type information 256 in database 236 to identify the communications protocol").

### **Claim 7**

Warren teaches a system comprising:

means for searching for at least one device based on a content type (Paragraph [0024] "A web services protocol is a protocol that may allow a web service to be published, located, and invoked over a network, although any other and/or additional functions may be supported by the web services protocol" and paragraph [0053] In one embodiment, database 236 stores device information 254. In a particular embodiment, device information 254 identifies each network element 108 in system 100 and the device type of each network element 108. For example, each network element 108 may communicate using at least one communications protocol, and network elements 108 may be divided into groups or device types based on the communications protocol used by network elements 108);

means for detecting the at least one device (See figure 4, number 404 "identify network element");

means for detecting a protocol associated with each device (See figure 4, number 406 "identify communications protocol");

means for matching the detected protocol with a protocol translator module; and  
means for using the protocol translator module to translate a command formatted in the



protocol into a translated command (See figure 4, number 414 “translate device command” and paragraph [0020] “In a particular embodiment, manager 102 communicates using a web services protocol, and abstraction device 106 translates between the web services protocol and the protocols used by network elements 108. This may allow manager 102 to communicate with different network elements 108 using a common protocol”).

Warren fails to teach wherein the translated command is formatted in a common application programming interface, wherein the common application programming interface is a single application programming interface that is configured to be used by a plurality of applications.

However, Lozinski teaches in Column 1 lines 50-54 “a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities” in order that “an application program can select at any particular time which of the facilities to communicate with using the interface” (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include “a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities” as taught by Lozinski in order that “an application program can select at any particular time which of the facilities to communicate with using the interface”.

### **Claim 8**

Warren teaches a method comprising:

searching for at least one service based on a content type (Paragraph [0024] "A web services protocol is a protocol that may allow a web service to be published, located, and invoked over a network, although any other and/or additional functions may be supported by the web services protocol" and paragraph [0053] In one embodiment, database 236 stores device information 254. In a particular embodiment, device information 254 identifies each network element 108 in system 100 and the device type of each network element 108. For example, each network element 108 may communicate using at least one communications protocol, and network elements 108 may be divided into groups or device types based on the communications protocol used by network elements 108);

detecting at the least one service (See figure 4, number 404 "identify network element");

detecting a protocol associated with each service (See figure 4, number 406 "identify communications protocol");

matching the detected protocol with a protocol translator module; and using the protocol translator module to translate a command formatted in the protocol into a translated command (See figure 4, number 414 "translate device command" and paragraph [0020] "In a particular embodiment, manager 102 communicates using a web services protocol, and abstraction device 106 translates between the web services

protocol and the protocols used by network elements 108. This may allow manager 102 to communicate with different network elements 108 using a common protocol").

Warren fails to teach wherein the translated command is formatted in a common application programming interface, wherein the common application programming interface is a single application programming interface that is configured to be used by a plurality of applications.

However, Lozinski teaches in Column 1 lines 50-54 "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" in order that "an application program can select at any particular time which of the facilities to communicate with using the interface" (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" as taught by Lozinski in order that "an application program can select at any particular time which of the facilities to communicate with using the interface".

### **Claim 9**

Warren teaches a method comprising:

searching for a specific device from a plurality of devices based on a content type (Paragraph [0024] "A web services protocol is a protocol that may allow a web service

to be published, located, and invoked over a network, although any other and/or additional functions may be supported by the web services protocol" and paragraph [0053] In one embodiment, database 236 stores device information 254. In a particular embodiment, device information 254 identifies each network element 108 in system 100 and the device type of each network element 108. For example, each network element 108 may communicate using at least one communications protocol, and network elements 108 may be divided into groups or device types based on the communications protocol used by network elements 108);

detecting the plurality of devices wherein each unique device communicates using a corresponding protocol (See figure 4, number 404 "identify network element");

displaying an indication of each device if a protocol translator module is matched with the corresponding protocol (See figure 4, number 406 and 412, "identify network element" and "Map information"); and

translating a command formatted in the corresponding protocol into a translated command (See figure 4, number 414 "translate device command" and paragraph [0020] "In a particular embodiment, manager 102 communicates using a web services protocol, and abstraction device 106 translates between the web services protocol and the protocols used by network elements 108. This may allow manager 102 to communicate with different network elements 108 using a common protocol").

Warren fails to teach wherein the translated command is formatted in a common application programming interface through the protocol translator module, wherein the

common application programming interface is a single application programming interface that is configured to be used by a plurality of applications.

However, Lozinski teaches in Column 1 lines 50-54 "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" in order that "an application program can select at any particular time which of the facilities to communicate with using the interface" (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" as taught by Lozinski in order that "an application program can select at any particular time which of the facilities to communicate with using the interface".

#### **Claim 10**

The modified Warren teaches the method according to claim 9, further comprising detecting the corresponding protocol from each device (See figure 4, number 406 "identify communications protocol").

#### **Claim 11**

The modified Warren teaches the method according to claim 9, further comprising storing the protocol translator module (paragraph [0052], "Database 236

may include any hardware, software, firmware, or combination thereof suitable to store and facilitate retrieval of information. Database 236 may store any suitable information used by abstraction device 206 to perform command translation or other functions" and paragraph [0054], "Database 236 may also store device type information 256. Device type information 256 may, for example, identify each device type in system 100, the communications protocol used by each of the device types, and any other suitable information about the device types").

#### **Claim 13**

The modified Warren teaches the method according to claim 9, further comprising searching for a specific device from the plurality of devices based on a device identifier (paragraph [0065] "Device identifier").

#### **Claim 15**

The modified Warren teaches the method according to claim 9, further comprising searching for a specific device from the plurality of devices based on a device type (paragraph [00703] "Device type identifier").

#### **Claim 16**

The modified Warren teaches the method according to claim 9, further comprising searching for a specific device from the plurality of devices based on a device's availability (paragraph [0065] "Device identifier" and "Other and/or additional

information may be included in identification information 352 without departing from the scope of the present invention”).

### **Claim 17**

Warren teaches a method comprising:

identifying a plurality of protocol translator modules wherein each protocol translator module is associated with a unique protocol; storing a list representing the plurality of protocol translator modules (paragraph [0052], “Database 236 may include any hardware, software, firmware, or combination thereof suitable to store and facilitate retrieval of information. Database 236 may store any suitable information used by abstraction device 206 to perform command translation or other functions” and paragraph [0054], “Database 236 may also store device type information 256. Device type information 256 may, for example, identify each device type in system 100, the communications protocol used by each of the device types, and any other suitable information about the device types”);

displaying an indication of each device having a device protocol that is compatible with one of the plurality of protocol translator modules in the list(See figure 4, number 406 “identify communications protocol”); and

translating a command formatted in the device protocol into a translated command (See figure 4, number 414 “translate device command” and paragraph [0020] “In a particular embodiment, manager 102 communicates using a web services protocol, and abstraction device 106 translates between the web services protocol and

the protocols used by network elements 108. This may allow manager 102 to communicate with different network elements 108 using a common protocol").

Warren fails to teach wherein the translated command is formatted in a common application programming interface through one of the plurality of protocol translator modules, wherein the common application programming interface is a single application programming interface that is configured to be used by a plurality of applications.

However, Lozinski teaches in Column 1 lines 50-54 "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" in order that "an application program can select at any particular time which of the facilities to communicate with using the interface" (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" as taught by Lozinski in order that "an application program can select at any particular time which of the facilities to communicate with using the interface".

### **Claim 18**

The modified Warren teaches the method according to claim 17, further comprising searching for additional protocol translator modules (paragraph [0052], "Database 236 may include any hardware, software, firmware, or combination thereof



suitable to store and facilitate retrieval of information. Database 236 may store any suitable information used by abstraction device 206 to perform command translation or other functions").

#### **Claim 19**

The modified Warren teaches the method according to claim 18, further comprising updating the index in response to the searching for additional protocol translator modules (paragraph [0052], "Database 236 may include any hardware, software, firmware, or combination thereof suitable to store and facilitate retrieval of information. Database 236 may store any suitable information used by abstraction device 206 to perform command translation or other functions").

#### **Claim 20**

Warren teaches a system comprising:

a first device configured for operating using a first protocol; a second device configured for operating using a second protocol; and a protocol translation layer configured for searching for a first protocol translation module corresponding to the first protocol and for searching for a second protocol translation module corresponding to the second protocol (paragraph [0006] "The apparatus further includes a plurality of protocol converters, each operable to receive at least one device command, translate the at least one device command from a first protocol to a second protocol, and communicate the at least one device command to one or more network or non-network

device elements. At least two of the protocol converters are operable to translate the at least one device command into different second protocols"), wherein the protocol translation layer is configured to translate a first command formatted in the first protocol into a command (See figure 4, number 414 "translate device command" and paragraph [0020] "In a particular embodiment, manager 102 communicates using a web services protocol, and abstraction device 106 translates between the web services protocol and the protocols used by network elements 108. This may allow manager 102 to communicate with different network elements 108 using a common protocol"), the first protocol translation module and second protocol translation module stored in a list representing a plurality of protocol translator modules ([0049] "Command translator 234 may receive commands 240 and translate and/or reformat commands 240 into one or more device commands 248 suitable for use by network elements 108. For example, command translator 234 may receive a command 240, access database 236 and/or an asset manager 246, and identify the network element 108 associated with command 240. Command translator 234 may also access database 236 and identify one or more device commands 248a and 248b (referred to collectively as device commands 248) that invoke the function requested by command 240 in the identified network element or elements 108. Command translator 234 may further use the information stored in database 236 to map information contained in command 240 into the appropriate positions or fields in device commands 248. In addition, command translator 234 may communicate each device command 248 to a protocol converter 238 that communicates with a network element 108 using the appropriate protocol" and

paragraph [0052] "Database 236 stores and facilitates retrieval of information used by abstraction device 206 to perform command translation and other functions. Database 236 may include any of a variety of data structures, arrangements, and/or compilations suitable to store and facilitate retrieval of information").

Warren fails to teach a plurality of applications configured for operating through a single, common application programming interface; wherein the translated command is formatted in the single, common application programming interface for use by one of the plurality of applications and to translate a second command formatted in the second protocol into a command formatted in the single, common application programming interface for use by another one of the plurality of applications.

However, Lozinski teaches in Column 1 lines 50-54 "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" in order that "an application program can select at any particular time which of the facilities to communicate with using the interface" (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" as taught by Lozinski in order that "an application program can select at any particular time which of the facilities to communicate with using the interface".

### **Claim 22**

The modified Warren teaches the system according to claim 20, further comprising a presentation layer configured for displaying the first device after locating the first protocol translation module (paragraph [0032] "Abstraction device 106 may include any hardware, software, firmware, or combination thereof for facilitating communication between components of system 100");

### **Claim 23**

Warren teaches a network protocol translation system comprising:

a processor that executes a plurality of run time processes (paragraph [0032] "Abstraction device 106 may include any hardware, software, firmware, or combination thereof for facilitating communication between components of system 100");

wherein the processor enables at least one of the run time processes to communicate via a first network protocol by executing a first translation module that translates between the first network protocol; and wherein the processor enables the at least one of the run time processes to communicate via a second network protocol, different from the first network protocol, by executing a second translation module that translates between the second network protocol and the application programming interface (See figure 4, number 414 "translate device command" and paragraph [0006] "The apparatus further includes a plurality of protocol converters, each operable to receive at least one device command, translate the at least one device command from a first protocol to a second protocol, and communicate the at least one device command

to one or more network or non-network device elements. At least two of the protocol converters are operable to translate the at least one device command into different second protocols”), further wherein the first translation module and second translation module are stored in a list representing a plurality of protocol translator modules ([0049] “Command translator 234 may receive commands 240 and translate and/or reformat commands 240 into one or more device commands 248 suitable for use by network elements 108. For example, command translator 234 may receive a command 240, access database 236 and/or an asset manager 246, and identify the network element 108 associated with command 240. Command translator 234 may also access database 236 and identify one or more device commands 248a and 248b (referred to collectively as device commands 248) that invoke the function requested by command 240 in the identified network element or elements 108. Command translator 234 may further use the information stored in database 236 to map information contained in command 240 into the appropriate positions or fields in device commands 248. In addition, command translator 234 may communicate each device command 248 to a protocol converter 238 that communicates with a network element 108 using the appropriate protocol” and paragraph [0052] “Database 236 stores and facilitates retrieval of information used by abstraction device 206 to perform command translation and other functions. Database 236 may include any of a variety of data structures, arrangements, and/or compilations suitable to store and facilitate retrieval of information”).

Warren fails to teach wherein the processes use only a single application programming interface for network communication; wherein the first translation module translates to a single application programming interface; and wherein the second translation module translates to a single application programming interface for network communication.

However, Lozinski teaches in Column 1 lines 50-54 "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" in order that "an application program can select at any particular time which of the facilities to communicate with using the interface" (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" as taught by Lozinski in order that "an application program can select at any particular time which of the facilities to communicate with using the interface".

#### **Claim 24**

Warren teaches a method, executed on a computing platform, comprising the acts of:

executing a plurality of run time processes (paragraph [0032] "Abstraction device 106 may include any hardware, software, firmware, or combination thereof for facilitating communication between components of system 100"); enabling at least one of the run time processes to communicate via a first network protocol by executing a first translation module that translates between the first network protocol; and enabling the at least one of the run time processes to communicate via a second network protocol, different from the first network protocol, by executing a second translation module that translates between the second network protocol (See figure 4, number 414 "translate device command" and paragraph [0006] "The apparatus further includes a plurality of protocol converters, each operable to receive at least one device command, translate the at least one device command from a first protocol to a second protocol, and communicate the at least one device command to one or more network or non-network device elements. At least two of the protocol converters are operable to translate the at least one device command into different second protocols"), wherein the first translation module and second translation module are stored in a list representing a plurality of protocol translator modules ([0049] "Command translator 234 may receive commands 240 and translate and/or reformat commands 240 into one or more device commands 248 suitable for use by network elements 108. For example, command translator 234 may receive a command 240, access database 236 and/or an asset manager 246, and identify the network element 108 associated with command 240. Command translator 234 may also access database 236 and identify one or more device commands 248a and 248b (referred to collectively as device commands 248)

that invoke the function requested by command 240 in the identified network element or elements 108. Command translator 234 may further use the information stored in database 236 to map information contained in command 240 into the appropriate positions or fields in device commands 248. In addition, command translator 234 may communicate each device command 248 to a protocol converter 238 that communicates with a network element 108 using the appropriate protocol" and paragraph [0052] "Database 236 stores and facilitates retrieval of information used by abstraction device 206 to perform command translation and other functions. Database 236 may include any of a variety of data structures, arrangements, and/or compilations suitable to store and facilitate retrieval of information").

Warren fails to teach wherein the processes use only a single application programming interface for network communication; wherein the first translation module translates to a single application programming interface; and wherein the second translation module translates to a single application programming interface for network communication.

However, Lozinski teaches in Column 1 lines 50-54 "a data processing system in which an application program can communicate with two or more system facilities via a programming interface common to the facilities" in order that "an application program can select at any particular time which of the facilities to communicate with using the interface" (Column 1 lines 56-58).

It would have been obvious to one of ordinary skill in the art at the time of the invention to create the invention of Warren to include "a data processing system in



which an application program can communicate with two or more system facilities via a programming interface common to the facilities” as taught by Lozinski in order that “an application program can select at any particular time which of the facilities to communicate with using the interface”.

**(10) Response to Argument**

The examiner summarizes the various points raised by the appellant and addresses them individually.

**(a) Claim 1: The appellant has argued that the modified Warren does not teach searching for at least one device based on a content type, detecting at least one device, detecting a protocol associated with the device, and matching the protocol with a protocol translator, and a single application programming interface that is configured to be used by a plurality of applications (Appeal Brief: page 14, paragraph 1). The appellant has referred back to an outline of arguments on pages 9-13 of the Appeal Brief in support of these statements (Appeal Brief: pages 9-13).**

**In response:**

The examiner respectfully disagrees for the following reasons:

In regards to searching for at least one device based on a content type, the appellant has presented arguments that “*warren teaches that its protocol translators are*

*all always loaded*" (Appeal Brief: page 10, paragraph 2) and *"the location and type of web service (content) are published and are therefore already known"* and *"Nothing in Warren teaches detecting anything by searching for it. To the contrary, Warren is silent about network elements which do not publish web services. Warren does not teach searching for at least one device based on a content type"* (Appeal Brief: page 11, paragraph 1) and *"Thus, the presently claimed invention creates a lightweight run-time binding by only loading the protocol translator module(s) which correspond to the at least one device found by virtue of being searched for based on its content"* (Appeal Brief: page 13 paragraph 2).

The examiner respectfully disagrees and although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). The examiner asserts that the claim does not limit "searching" in the manner argued by the appellant and furthermore, none of the claims contain any limitations regarding loading of the module(s). The claim recites searching for at least one device based on a content type which the examiner will show is within the scope of Warren.

Paragraph [0024] of Warren teaches "A web services protocol is a protocol that may allow a web service to be published, located, and invoked over a network, although any other and/or additional functions may be supported by the web services protocol" and "In this embodiment, a network element 108 may be treated as a web service. In this manner, functions of the network element 108 may be published, located, and invoked over network 104. In this embodiment, the web services protocol may describe

how to access and invoke functions of the network element 108". Furthermore in paragraph [0053] Warren teaches "In one embodiment, database 236 stores device information 254. In a particular embodiment, device information 254 identifies each network element 108 in system 100 and the device type of each network element 108. For example, each network element 108 may communicate using at least one communications protocol, and network elements 108 may be divided into groups or device types based on the communications protocol used by network elements 108". The examiner asserts that the appellant's searching for a device based on a content type reads upon Warren's locating a network element wherein network elements may be divided into groups or device types based on communication protocols.

In regards to detecting at least one device, detecting a protocol associated with the device, and matching the protocol with a protocol translator, the appellant has presented arguments similar to those in regards to searching for at least one device based on a content type. Specifically the appellant argues, *"Further, as discussed above, Warren teaches that its protocol translators are all always loaded. Warren does not teach matching the detected protocol with a protocol translator"* (Appeal Brief: page 10, paragraph 2).

The examiner respectfully disagrees for similar reasoning presented above in regards to searching for at least one device based on a content type. The examiner asserts that the claim does not limit "matching" in the manner argued by the appellant and furthermore, none of the claims contain any limitations regarding loading of the

module(s). The claim recites detecting the at least one device, detecting a protocol associated with each device and matching the detected protocol with a protocol translator which the examiner will show is within the scope of Warren.

In regards to detecting at least one device and detecting a protocol associated with the device, Warren teaches in paragraph [0070] "Abstraction device 206 identifies one or more network elements 108 associated with the command 240 at step 404" and "Abstraction device 206 identifies the communications protocol used by the identified network element 108 at step 406. This may include, for example, command translator 234 using device information 254 and/or device type information 256 in database 236 to identify the communications protocol". In regards to matching the protocol with a protocol translator, Warren teaches in paragraph [0074] "Abstraction device 206 translates the device command 248 at step 414. This may include, for example, command translator 234 communicating the device command 248 to one of the protocol converters 238. The protocol converter 238 selected may be based, for example, on the communications protocol identified at step 406. This may also include command translator 234 communicating the device command 248 to the protocol converter 238 using a first protocol and the protocol converter 238 translating the device command 248 into a second protocol used by network element 108". A general outline of these steps as described above (404, 406 and 414) can be found in Figure 4 of Warren.

In regards to a single application programming interface that is configured to be used by a plurality of applications, the appellant has presented arguments that "*Lozinski*

*is cited as teaching wherein the common application programming interface is a single application programming interface that is configured to be used by a plurality of applications. [Office Action of June 22, 2009, pages 3-4] The Applicants respectfully disagree. As described above, Lozinski teaches one application accessing a plurality of similar devices using a common programming interface. [Lozinski, Fig. 1, 120]. Lozinski does not teach a single programming interface that is configured to be used by a plurality of applications" (Appeal Brief: page 12, paragraph 2).*

The examiner respectfully disagrees and assumes the appellant is referring to Figure 2 of Lozinski as there is no item "120" in Figure 1. Lozinski teaches in Column 2 line 65 - Column 3 line 8, "In this embodiment of the invention, application programs can communicate across the communications networks via communications adapters 90, 100, 110 using a programming interface to communicate with programs controlling the adapters. This relationship is illustrated in FIG. 2. Application program 120 communicates with programs 130, 140 and 150 which enable communications via the adapters 90, 100 and 110 respectively. Programs 130, 140 and 150 communicate with application program 120 using a common programming interface. This programming interface is represented in FIG. 2 by the arrows 160".

The examiner asserts that Lozinski teaches wherein the common application programming interface is a single application programming interface (Lozinski Fig. 2: application program 120; and common program interface 160) to communicate with a plurality of applications (Lozinski Fig. 2: programs 130, 140, 150).

**(b) Claims 2 and 4-6:** The appellant has argued, “*Claims 2 and 4-6 are all dependent on the independent Claim 1. As described above, the independent Claim 1 is allowable over the teachings of Warren, Lozinski and their combination. Accordingly, Claims 2 and 4-6 are all also allowable as being dependent on an allowable base claim*” (Appeal Brief: page 14, paragraph 2).

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 1.

**(c) Claim 7:** The appellant has argued, “*As discussed above, Warren, Lozinski, and their combination do not teach searching for at least one device based on a content type. As also discussed above, Warren, Lozinski, and their combination, do not teach detecting a protocol associated with the device, and matching the protocol with a protocol translator. Further, as discussed above, Warren, Lozinski, and their combination, do not teach a single application programming interface that is configured to be used by a plurality of applications. For at least these reasons, the independent Claim 7 is allowable over the teachings of Warren, Lozinski and their combination*” (Appeal Brief: page 15, paragraph 1).

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 1, as the arguments regarding claim 7 are analogous in scope.

**(d) Claim 8: The appellant has argued, “As discussed above, Warren, Lozinski, and their combination do not teach searching for at least one device based on a content type. As also discussed above, Warren, Lozinski, and their combination, do not teach detecting at least one device, detecting a protocol associated with the device and matching the protocol with a protocol translator. Further, as discussed above, Warren, Lozinski, and their combination, do not teach a single application programming interface that is configured to be used by a plurality of applications. For at least these reasons, the independent Claim 8 is allowable over the teachings of Warren, Lozinski and their combination” (Appeal Brief: page 15, paragraph 2).**

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 1, as the arguments regarding claim 8 are analogous in scope.

**(e) Claim 9:** The appellant has argued, “*As discussed above, Warren, Lozinski, and their combination do not teach searching for at least one device based on a content type. As also discussed above, Warren, Lozinski, and their combination, do not teach detecting at least one device, detecting a protocol associated with the device and matching the protocol with a protocol translator. Further, as discussed above, Warren, Lozinski, and their combination, do not teach a single application programming interface that is configured to be used by a plurality of applications. For at least these reasons, the independent Claim 9 is allowable over the teachings of Warren, Lozinski and their combination*” (Appeal Brief: page 16, paragraph 1).

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 1, as the arguments regarding claim 9 are analogous in scope.

**(f) Claims 10, 11, 13, 15 and 16:** The appellant has argued, “*Claims 10, 11, 13, 15 and 16 are all dependent on the independent Claim 9. As described above, the independent Claim 9 is allowable over the teachings of Warren, Lozinski and their combination. Accordingly, Claims 10, 11, 13, 15 and 16 are all also allowable as being dependent on an allowable base claim*” (Appeal Brief: page 16, paragraph 2).



**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 9.

**(g) Claim 17: The appellant has argued, “As discussed above, Warren, Lozinski, and their combination, do not teach storing a list of network protocols available for use and displaying an indication of each device having a device protocol that is compatible with one of the plurality of protocol translator modules. Further, as discussed above, Warren, Lozinski, and their combination, do not teach a single application programming interface that is configured to be used by a plurality of applications. For at least these reasons, the independent Claim 17 is allowable over the teachings of Warren, Lozinski and their combination” (Appeal Brief: page 17, paragraph 1).**

**In response:**

The examiner respectfully disagrees.

In regards to storing a list of network protocols available for use, Warren teaches in paragraph [0052], “Database 236 stores and facilitates retrieval of information used by abstraction device 206 to perform command translation and other functions. Database 236 may include any of a variety of data structures, arrangements, and/or

compilations suitable to store and facilitate retrieval of information" and paragraph [0054], "Database 236 may also store device type information 256. Device type information 256 may, for example, identify each device type in system 100, the communications protocol used by each of the device types, and any other suitable information about the device types".

In regards to displaying an indication of each device having a device protocol that is compatible with one of the plurality of protocol translator modules, Warren teaches in paragraph [0070] "Abstraction device 206 identifies one or more network elements 108 associated with the command 240 at step 404. This may include, for example, command translator 234 and/or asset manager 246 using identification information 352 in command 240 to identify the one or more network elements 108. Abstraction device 206 identifies the communications protocol used by the identified network element 108 at step 406. This may include, for example, command translator 234 using device information 254 and/or device type information 256 in database 236 to identify the communications protocol". Furthermore, Warren teaches supporting multiple types of protocols in paragraph [0072], "Abstraction device 206 identifies the format of the one or more device commands 248 at step 410. This may include, for example, command translator 234 accessing control instruction information 260 and/or script instruction information 262 in database 236. If a network element 108 receiving a device command 248 has a programmatic interface 112, such as a CORBA, SOAP, or SNMP interface 112a-112c, command translator 234 may use control instruction information 260. If a network element 108 receiving a device command 248 uses a command line interface

112d, command translator 234 may use script instruction information 262". The examiner asserts that Warren's teachings of identifying the communications protocol used by the network element and translating device commands constitutes an indication of the device having a device protocol that is compatible with one of the plurality of protocol translator modules.

**(h) Claims 18 and 19: The appellant has argued, "*Claims 18 and 19 are both dependent on the independent Claim 17. As described above, the independent Claim 17 is allowable over the teachings of Warren, Lozinski and their combination. Accordingly, Claims 18 and 19 are both also allowable as being dependent on an allowable base claim*" (Appeal Brief: page 17, paragraph 2).**

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 17.

**(i) Claim 20: The appellant has argued, "As discussed above, Warren, Lozinski, and their combination, do not teach storing a list of network protocols available for use and displaying an indication of each device having a device protocol that is compatible with one of the plurality of protocol translator**

**modules. Further, as discussed above, Warren, Lozinski, and their combination, do not teach a single application programming interface that is configured to be used by a plurality of applications. For at least these reasons, the independent Claim 20 is allowable over the teachings of Warren, Lozinski and their combination” (Appeal Brief: page 18, paragraph 1).**

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 17, as the arguments regarding claim 20 are analogous in scope.

**(j) Claim 22: The appellant has argued, “*Claim 22 is dependent on the independent Claim 20. As described above, the independent Claim 20 is allowable over the teachings of Warren, Lozinski and their combination. Accordingly, Claim 22 is also allowable as being dependent on an allowable base claim*” (Appeal Brief: page 18, paragraph 2).**

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 20.

**(k) Claim 23: The appellant has argued, “As discussed above, Warren, Lozinski, and their combination, do not teach storing a list of network protocols available for use and displaying an indication of each device having a device protocol that is compatible with one of the plurality of protocol translator modules. Further, as discussed above, Warren, Lozinski, and their combination, do not teach a single application programming interface that is configured to be used by a plurality of applications. For at least these reasons, the independent Claim 23 is allowable over the teachings of Warren, Lozinski and their combination” (Appeal Brief: page 19, paragraph 1).**

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 17, as the arguments regarding claim 23 are analogous in scope.

**(l) Claim 24: The appellant has argued, “As discussed above, Warren, Lozinski, and their combination do not teach executing a plurality of run time processes that uses only a single application programming interface for network communication. For at least these reasons, the independent Claim 24 is allowable over the teachings of Warren, Lozinski and their combination” (Appeal Brief: page 20, paragraph 1).**

**In response:**

The examiner respectfully disagrees for reasons analogous to those provided in response to claim 17, as the arguments regarding claim 24 are analogous in scope.

**(11) Related Proceeding(s) Appendix**

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Farhad Ali/

Examiner, Art Unit 2446

/Jeffrey Pwu/

Supervisory Patent Examiner, Art Unit 2446

Conferees:

/Jeffrey Pwu/

Supervisory Patent Examiner, Art Unit 2446

Art Unit: 2446

/Benjamin R Bruckart/

Primary Examiner, Art Unit 2446